

## ASSIGNMENT 4

Student Name	Balaji.V
Student Roll Number	820519205009
Maximum Marks	2 Marks
Project Name	Containment Zone Alerting Application
Team ID	PNT2022TMID52453

### Questions:

1. Pull an image from docker hub and run it in docker playground.
2. Create a docker file for the jobportal application and deploy it in Docker desktop application.
3. Create a docker file for the jobportal application and deploy it in Docker desktop application.
4. Create a IBM container registry and deploy helloworld app or jobportal app.
5. Create a Kubernetes cluster in IBM cloud and deploy helloworld image or jobportal image and also expose the same app to run in nodeport

### Output:

#### Pull image in docker:

```
root@admin:/home/ash/Desktop... root@admin:/home/ash/Desktop... root@admin:/home/ash/Desktop... root@admin:/home/ash/Desktop... root@admin:/home/ash/Desktop...
root@admin:/home/ash/Desktop/Flask ClassWorrk# docker pull icr.io/helloind/python:1.0
root@admin:/home/ash/Desktop/Flask ClassWorrk# docker push icr.io/helloind/python:1.0
The push refers to repository [icr.io/helloind/python]
f7a508b1c151: Pushed
>7b353df37a8: Pushing [=====] 12.38MB
7216ce2329ba: Pushed
>f61ccc2347a: Pushed
>fc1deb8136e: Pushed
1f123186824c: Pushed
3d6eb1152931: Pushing [=====>] 7.748MB/57.12MB
100796cdf3b1: Pushing [=====] 11.14MB/18.48MB
>4acb5a6fa0b: Pushing [=>] 4.294MB/528.7MB
>3d51c618126f: Pushing [=>] 1.063MB/152MB
>ff6e4d46744: Waiting
>89d1d47b5a1: Waiting
>55ed1b7a428: Waiting
root@admin:/home/ash/Desktop/Flask ClassWorrk# docker push icr.io/helloind/python:1.0
The push refers to repository [icr.io/helloind/python]
f7a508b1c151: Layer already exists
>7b353df37a8: Layer already exists
7216ce2329ba: Layer already exists
>f61ccc2347a: Layer already exists
>fc1deb8136e: Layer already exists
1f123186824c: Layer already exists
3d6eb1152931: Pushed
100796cdf3b1: Pushed
>4acb5a6fa0b: Pushed
>3d51c618126f: Pushed
>ff6e4d46744: Pushed
>89d1d47b5a1: Pushed
>55ed1b7a428: Pushed
1.0: digest: sha256:9d7b9f369391c18d6aa88270ebb467ab6b93b3f18e61a46c9467b7fa5cc4b404 size: 3050
```

## Run Docker:

```
Login Succeeded

Logging in with your password grants your terminal complete access to your account.
For better security, log in with a limited-privilege personal access token. Learn more at https://docs.docker.com/go/ac
root@admin:/home/ash/Desktop/Flask ClassWorrk# docker run -p5000:5000 flakapp:1.0
Unable to find image 'flakapp:1.0' locally
docker: Error response from daemon: pull access denied for flakapp, repository does not exist or may require 'docker lo
requested access to the resource is denied.
See 'docker run --help'.
root@admin:/home/ash/Desktop/Flask ClassWorrk# docker run -p5000:5000 -t flakapp:1.0
Unable to find image 'flakapp:1.0' locally
docker: Error response from daemon: pull access denied for flakapp, repository does not exist or may require 'docker lo
requested access to the resource is denied.
See 'docker run --help'.
root@admin:/home/ash/Desktop/Flask ClassWorrk# docker run -p5000:5000 -t flakapp
Unable to find image 'flakapp:latest' locally
docker: Error response from daemon: pull access denied for flakapp, repository does not exist or may require 'docker lo
requested access to the resource is denied.
See 'docker run --help'.
root@admin:/home/ash/Desktop/Flask ClassWorrk# docker run -p5000:5000 -t flaskapp:1.0
docker: Error response from daemon: driver failed programming external connectivity on endpoint silly_bardeen (717385e0
d59728f1e3577afe801b18edb0257c07b377889d): Bind for 0.0.0.0:5000 failed: port is already allocated.
ERRO[0002] error waiting for container: context canceled
root@admin:/home/ash/Desktop/Flask ClassWorrk# docker run -t flaskapp:1.0
* Serving Flask app 'app'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://172.17.0.3:5000
Press CTRL+C to quit
172.17.0.1 - - [23/Oct/2022 17:21:29] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [23/Oct/2022 17:21:29] "GET /favicon.ico HTTP/1.1" 404 -
172.17.0.1 - - [23/Oct/2022 17:21:50] "GET / HTTP/1.1" 200 -
```

## Docker Playground:

labs.play-with-docker.com/p/cdagodu0qau0009n3080#cdagodu0\_cdagolu0qau0009n309g

03:56:24

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.8  
node1

cdagodu0\_cdagolu0qau0009n309g



IP  
192.168.0.8  
OPEN PORT

Memory  
SSH  
ssh ip172-18-0-20-cdagodu0qau0009n3080@direct.labs.play-with-  
DELETE EDITOR

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
00183bfe1385	mongo	"docker-entrypoint.s..."	15 seconds ago	Up 13 seconds	27017/tcp, 0.0
tcp_mongodb					
[node1] (local) root@192.168.0.8 ~					
\$ docker start mongodb					
mongodb					
[node1] (local) root@192.168.0.8 ~					
\$ docker images					
REPOSITORY	TAG	IMAGE ID	CREATED	SIZE	
mongo	latest	1cca5cf68239	2 weeks ago	695MB	
[node1] (local) root@192.168.0.8 ~					
\$ docker ps -a					
CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
00183bfe1385	mongo	"docker-entrypoint.s..."	About a minute ago	Up About a minute	27017/
0->3001/tcp_mongodb					
[node1] (local) root@192.168.0.8 ~					

03:53:15

CLOSE SESSION

Instances  

+ ADD NEW INSTANCE

192.168.0.8  
node1

cdagodu0\_cdagolu0qau0009n309g

IP

192.168.0.8

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-20-cdagodu0qau0009n3080@direct.labs.play-with-docker.com

DELETE

EDITOR

```

$ docker run -d -p6010:3000 hello-world
9b7c8f4adb31df3a7ceaf431f63e2eae868a579059402bc72b630b7e5e911875
^[[D[node1] (local) root@192.168.0.8 ~
$ docker run -d -p6020:3000 hello-world
6b0f1699a8ced253d9df780a42eeff298a0c2fb2c6389e709681da2e63879b0b
[node1] (local) root@192.168.0.8 ~
$ docker run -p6120:3000 hello-world

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)

```

## Container Registry:

cloud.ibm.com/registry/repos

Document Hoppscotch ... New Tab React Dialog ... GitHub Using Python ... How to MAKE...

IBM Cloud

Search resources and products...

Catalog Manage ASWIN s's Account

Container Registry

Quick start

Namespaces 1

Repositories 2

Images 2

Trash 0

Settings

Repositories

Location Global

Search

Create +

<input type="checkbox"/>	Name	Image count	Namespace	Last updated
<input checked="" type="checkbox"/>	python icr.io/helloind/python	1	helloind	11 hours ago
<input checked="" type="checkbox"/>	app icr.io/helloind/app	1	helloind	395 days ago

Items per page: 25

1-2 of 2 items

1 1 of 1 page



## KUBERNETES:

### dashboard-adminuser.yaml

apiVersion: v1

kind: ServiceAccount

metadata:

name: admin-user

namespace: kubernetes-dashboard

---

apiVersion: v1

kind: Secret

metadata:

name: admin-user-token

namespace: kubernetes-dashboard

annotations:

kubernetes.io/service-account.name: admin-user

type: kubernetes.io/service-account-token

---

apiVersion: rbac.authorization.k8s.io/v1

kind: ClusterRoleBinding

metadata:

name: admin-user

roleRef:

apiGroup: rbac.authorization.k8s.io

kind: ClusterRole

name: cluster-admin

subjects:

- kind: ServiceAccount

name: admin-user

namespace: kubernetes-dashboard

**Flask-deployment.yaml**

apiVersion: apps/v1

kind: Deployment

metadata:

name: flask-app

spec:

replicas: 3

selector:

matchLabels:

app: flask-app

template:

metadata:

labels:

app: flask-app

spec:

containers:

- name: flask-app-container

image: flask-app-testing

imagePullPolicy: Never

ports:

- containerPort: 5000

protocol: TCP

**Flask\_ingress.yaml:**

apiVersion: networking.k8s.io/v1

kind: Ingress

metadata:

name: flask-app-ingress

annotations:

kubernetes.io/ingress.class: nginx

nginx.ingress.kubernetes.io/ssl-redirect: "false"

spec:

# ingressClassName: nginx

rules:

- http:

paths:

backend:

service:

name: flask-app-service

port:

```
      number: 5000
    path: /
    pathType: Prefix
```

**Flask\_service.yaml:**

```
apiVersion: v1
kind: Service
metadata:
  name: flask-app-service
spec:
  type: ClusterIP
  ports:
    - port: 5000
  selector:
    app: flask-app
```